

**ABSTRACT OF THE DISCLOSURE**

A method and system is provided for migrating processes from one virtual machine to another on a network. To migrate the external state of a process, the process may use a network service connection system or a compact network service connection system for accessing resources external to the virtual machine. A process may be migratable separately from other processes. A process may have an in-memory heap used for the execution of the process, a virtual heap that may include the entire heap of the process including at least a portion of the runtime environment, and a persistent heap where the virtual heap may be checkpointed. In one embodiment, the virtual heap may serve as the persistent heap. In another embodiment, the virtual heap may be checkpointed to a separate, distinct persistent heap. The combination of the in-memory heap, the virtual heap, and the persistent store may be referred to as a virtual persistent heap. One embodiment of a method for migrating an application may include checkpointing the application to a persistent heap. Current leases to local and/or remote resources may be expired. The persistent state of the process may be packaged in the persistent heap and sent to the node where the process is to migrate. A transaction mechanism may be used, where the process's persistent state is copied and committed as having migrated on both the sending and receiving nodes. The state of the process may then be reconstituted into a new virtual persistent heap on the node where the application migrated. Leases to local and/or remote resources for the process may be re-established. The process may then resume execution on the node where it migrated. In one embodiment, a versioning mechanism may be used whereby nodes where a process once lived may cache a previous state. In addition, a user interface (UI) may be provided to manage process checkpoints.